DOCUMENT RESUME

ED 082 980 SE 016 699

AUTHOR	Blaylock, James E.; Jones, Lonnie L.
TITLE	Economic and Ecological Input-Output Model.
INSTITUTION	Texas A and M Univ., College Station. Texas Agricultural Experiment Station.
PUB DATE	73
NOTE	55p.
EDRS PRICE	MF-$0.65 HC-$3.29
DESCRIPTORS	*Computer Programs; *Economics; *Employment; Environment; Income; *Input Output; *Models; Regional Planning
IDENTIFIERS	FORTRAN IV

ABSTRACT

This documentation presents an input-output model which has been modified to include the environmental impact of economic operation. In lieu of market prices for the environmental factors, trade-offs with regional income and employment are estimated for use in regional planning. The program is written in FORTRAN IV with single precision for the IBM 360/65 system. The example data has been set-up to contain five endogenous sectors and three exogenous sectors (including households). The data are then compressed to three endogenous sections. (Author/BL)

FILMED FROM BEST AVAILABLE COPY

ED 082980

Department of Agricultural Economics and Rural Sociology 1973

U.S. DEPARTMENT OF HEALTH, EDUCATION & WELFARE
NATIONAL INSTITUTE OF EDUCATION
THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY.

Departmental Program and Model Documentation

ECONOMIC AND ECOLOGICAL INPUT-OUTPUT MODEL

73-2

The Texas A&M University System
The Texas Agricultural Experiment Station
J. E. Miller, Director, College Station

SE 016 699

ED 082980

Economic and Ecological Input-Output Model

Agricultural Economics
Program and Model Documentation
73-2

by

James E. Blaylock

and

Lonnie L. Jones

I. PROGRAM IDENTIFICATION AND BACKGROUND

1. Program Name . . . . . Economic and Ecological Input-Output Model
2. Documentation Number 73-2
3. Author of Documentation. . . . . J.E. Blaylock and L.L. Jones
4. Programmer . . . . . Don Book and J.E. Blaylock
5. Origin of Program. . . . . Original
6. Language/Computer. . . . . Fortran IV, IBM 360/65
7. Date . . . . . June, 1973

Page Number

II. GENERAL ABSTRACT. 1

III. PURPOSE AND GENERAL CAPABILITIES OF THE PROGRAM 2

IV. USER DOCUMENTATION. 4

1. Data Requirements 4
2. Generalized Job Stream. 6
3. Data Coding 8
4. Example Code Sheets 14
5. Complete Job Stream 17
6. Output. 17
7. Example Problem 19

V. PROGRAMMER DOCUMENTATION. 26

1. Basic Model 26
2. Calculation of Multipliers. 29

VI. PROGRAM DESCRIPTION 32

1. Routines. 32
2. Dimensions and Initial Data Statements. 33
3. Flow Chart 35
4. Program Listing 37

VII. BIBLIOGRAPHY. 52

General Abstract

This documentation presents an input-output model which has been modified to include the environmental impact of economic operation. In lieu of market prices for the environmental factors, trade-offs with regional income and employment are estimated for use in regional planning. The program is written in FORTRAN IV with single precision for the IBM 360/65 system. The example data has been set-up to contain five endogenous sectors and three exogenous sectors (including households). The data is then compressed to three endogenous sectors.

PURPOSE AND GENERAL CAPABILITIES OF THE PROGRAM

Input-Output is a well known analytical tool which is particularly well suited for gaining information on the economic aspects of environmental quality and resource use. These economic aspects concern trade-offs between levels of pollution or resource use and levels of output, income, and employment in the economy. When an environmental sector is added to the I-O matrix, these trade-offs may be estimated by using interdependence coefficients and the multiplier concept normally incorporated in input-output models. The purpose of this documentation is to provide information about a computer program (ECON-ECOL) that allows an analysis of economic and environmental data via the input-output approach by estimating interrelationships and multipliers which link the economy with its demand for resources and its supply of pollutants.

The details of the program's operation are described in later sections, but a statement of the general aspects of the program is useful as an introduction. The program links the environmental factors with the economic sectors by expressing resources as inputs per dollar of output and pollutants as production by-products per dollar of output. Environmental factors are introduced into the input-output model as L sectors of positive environmental imports (resource inputs) and K sectors of negative environmental imports (exported pollutants). The result is a MXN (where M = K + L) environmental matrix. The data in this matrix are converted to quantities per unit of output (production coefficients) for each economic sector. This yields a more complete production function for each economic sector of the model.

Once formed, the environmental coefficient matrix is post multiplied

by the inverse of the economic processing matrix. The resulting matrix contains environmental-output (E-0) multipliers that indicate the direct and indirect effects of economic production on the environment. By using the

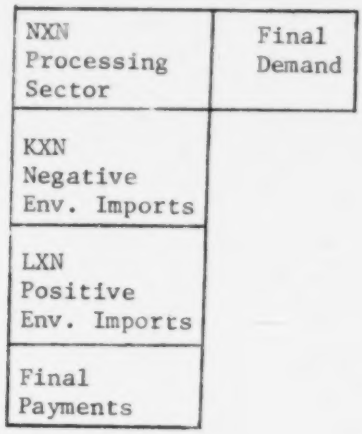


FIGURE 1

inverse of a processing sector containing households, the induced environmental effect is incorporated in the E-0 multipliers. Finally, by combining the elements of the E-0 matrix with appropriate income and employment data, environmental trade-offs in terms of dollars of income and/or levels of employment are generated.

The program is written to provide maximum information and flexibility and to be relatively inexpensive with respect to computer operation. The information provided as output includes multipliers calculated with households both exogenous (type I) and endogenous (type II). The flexibility provides for any size matrix up to 95 x 95 for the complete I-0 model, up to 90 x 90 for the processing sector, and up to 25 environmental factors. The environmental factors can be either pollutants or resources or both. A compression subroutine allows a large model to be aggregated to as small a matrix as is desired. Other options allow for deleting the employment vectors or for operating simply as an I-0 model excluding the environmental analysis.

USER DOCUMENTATION

The use of the various options of the ECON-ECOL program results in differing data requirements, arrangements, and coding. The use of the option COMPRESS calls for a slight alteration in the way in which information is presented to the computer. Information on data requirements and the arrangement of data relative to the main program is presented here as if the program's full capabilities are to be used.

Data Requirements

Assuming that an analysis including employment-environmental relationships is desired, data is needed for two matrices and two vectors. The matrices are for economic data on each industry's sales and purchases and environmental data on resources required and pollutants produced by each industry listed in the processing sector of the I-O model. The two vectors consist of employment totals for each industry and a list of names for the industries and environmental factors that constitute the production function.

Economic Data

It will be assumed that the user is familiar with the data requirements of the economic sector of this input-output model. Therefore, the only discussion of this topic will center on level of aggregation for each sector. This topic will be taken up later.

Environmental Data

The environmental data must contain environmental inputs for each industry of the processing sector expressed as positive quantities, and

pollutants expressed as negative quantities. The number and type of these positive and negative environmental factors depends on existing definitions of pollutants, data availability, and the user's analytical interests. The environmental data should conform to the economic data in terms of industrial classification, production period, etc. However, unlike the economic data that is expressed in dollars, resources and/or pollutants may be expressed in any appropriate units (e.g. tons, gallons, acre feet). Differing units may be used within the same environmental matrix.

Employment Data

Employment totals for each of the industries within the processing sector are used to calculate environmental-employment trade-offs as well as the standard I-O employment multipliers.

Row Names

For print-out purposes row names for the economic and environmental matrices are needed. The economic row names include processing industries, households, and final purchases. Names for final demand sectors are not needed as they are listed as columns which are numbered rather than named. Environmental row names contain designations for each resource and each pollutant for which data has been collected.

Each of the above data items may be thought of as separate data decks in the program. Hereafter, they are referred to as ECON for the economic data, RES (resources) for the positive and negative environmental data, EMPLOY for employment, and NAME for row name vector.

Control Cards

The final data category contains control and option cards. The options desired are listed on one computer card, called the option card (designated

OPT). The options available are for environmental analysis, employment-environmental analysis, and compression of original data. If the environmental analysis is excluded, resource and pollutant data is not needed, and the program will perform a standard Input-Output analysis. This analysis results in output multipliers, type I & II income multipliers, and if desired, employment multipliers. Employment multipliers for the economic sectors and the environmental sectors can be excluded via the second option. Through the compression option it is possible to reduce the matrix constructed from the original data collection to as small a size as is desirable. The result will be a reduced economic and resource matrix with a correspondingly shortened employment vector.

The control card SIZE specifies the size and degree of partitioning of the matrix that is to be used in the analysis for that particular computer run. The size of the matrix refers to the entire transaction table. The partitioning consists of separating and finding subtotals for the processing matrix, the value added vectors, and the import vectors.

The final set of control cards are used only if subroutine compress is called. These cards describe which rows and columns are to be aggregated and the position of the resulting rows and columns in the reduced matrix. These control cards, then, constitute another data "deck" and will be designated MOVE.

Generalized Job Stream

The control card, option card, and data decks are combined with the main program as illustrated in Figure 2.

GENERAL JOB STREAM

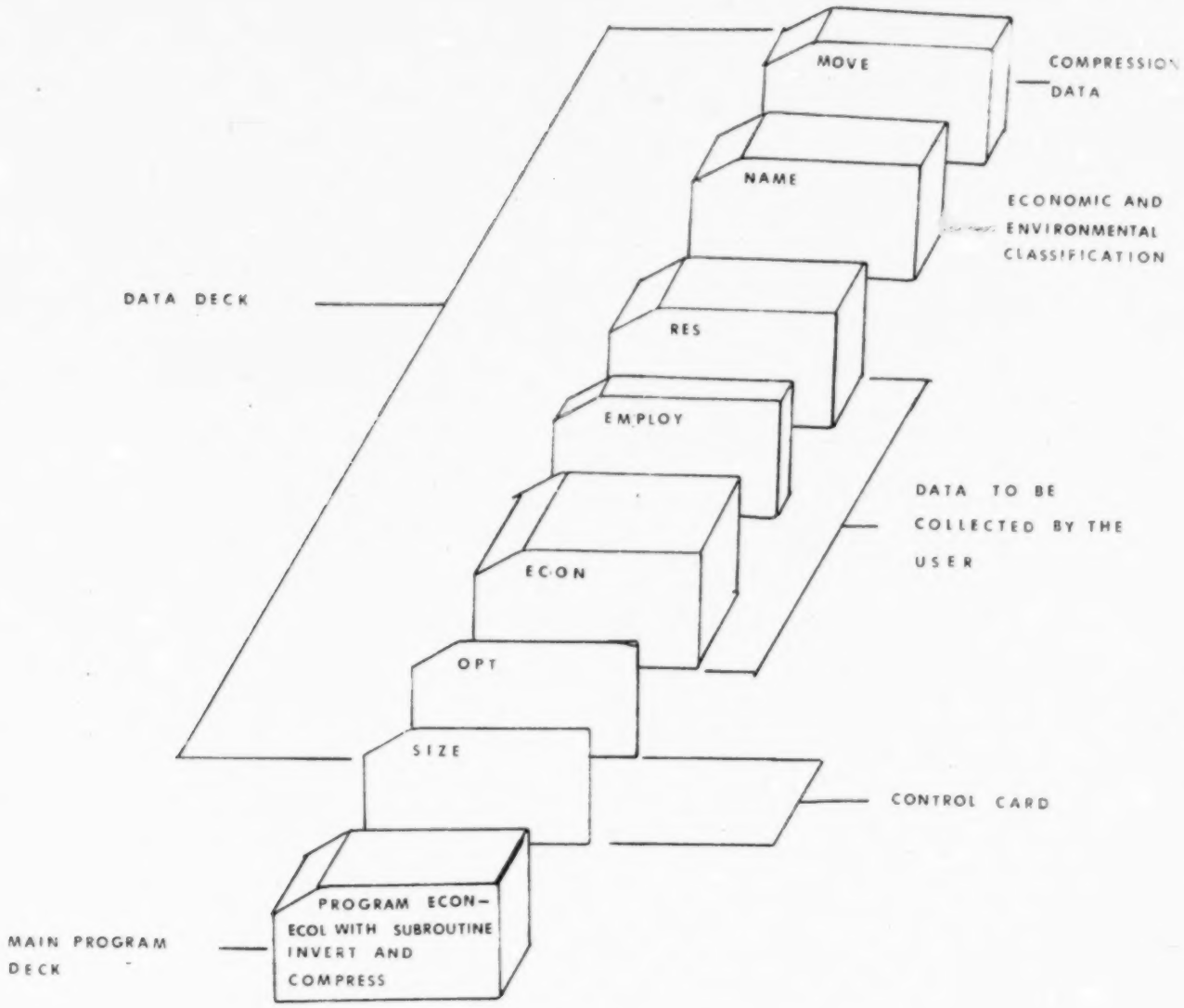


FIGURE 2

Data Coding

Industrial Classification

The level of aggregation for which the data is collected depends on the industry classification. This classification also determines the size of the matrices to be used. This program is written in general terms so that any size matrices can be used that are less than or equal to the maximum specified in the dimension statements. These limits consist of 95 x 95 for the I-O transaction table, 89 x 89 for the processing sector without households, and 25 x 100 for the resource matrix. Larger or smaller limits can be used but it will be necessary to change several cards in the program. This information can be found in the section describing the technical details of the program itself.

Economic and Resource Format

The collected data must be coded so that they can be read by the computer. The economic and environmental data are presented in the same form. Each quantity to be placed in a matrix is accompanied by the row and column number which designates its position in the matrix. Each data card will have four data points. Each data point will consist of three numbers. These numbers are row, column, and data value. The exact format is specifed as 4(2I3, F12.3).

	Card Item	Card Column	Format
1)	Row number	1-3	I3
2)	Column number	4-6	I3
3)	Datum value	7-18	F12.3
4)	Row number	19-21	I3
5)	Column number	22-24	I3

Card Item	Card Column	Format
6) Datum value	25-36	F12.3
⋮	⋮	⋮
10) Row number	55-57	I3
11) Column number	58-60	I3
12) Datum value	61-72	F12.3

This arrangement of data is repeated on as many additional cards as needed.

Data Arrangement

Zeroes need not be coded since all matrices are filled with zeroes initially and whatever cells are not filled with data will automatically register as zero. An additional aspect of this feature is that the order of the data within the decks ECON & RES is unimportant. Each deck could be shuffled and the row and column designation will insure proper placement of the quantity.

There are only two restrictions on the arrangement of the data. First, the households sector must be the vector immediately following the processing matrix. This facilitates bringing households inside the processing sector for estimating type II multipliers. It is, therefore, necessary for the household row and column total to be balanced. Second, all value added sectors must be grouped together and preceed a similar grouping of the import sectors. These groupings are necessary because of the feature that partitions the transaction table and presents row sub-totals for each of the groups mentioned.

Pollutants are treated as negative resources. Therefore, each pollutant quantity should be punched as a negative number. The last card in deck ECON and deck RES must be -99 starting in column one. This tells the computer that the last card in that deck has been read.

Employment Format

Employment totals are placed six to a card. Each figure is right

justified in a ten space field beginning in column one. As many cards as are necessary may be used.

Item	Card Column	Format
Card 1		
1. Employment total sector 1	1-10	I10
2. Employment total sector 2	11-20	I10
⋮	⋮	⋮
6. Employment total sector 6	51-60	I10
Card 2		
1. Employment total sector 7	1-10	I10
2. Employment total sector 8	11-20	I10
⋮	⋮	⋮
6. Employment total sector 12	51-60	I10

Row Name Format

Row names are punched one to a card. The first three spaces contain the row number. The next three spaces are left blank, and the names are then placed anywhere within a twelve character field. This is done for both economic sectors and environmental categories.

Item	Card Column	Format
Card 1		
1. Row number	1-3	I3
2. Row name	7-18	3A4
Card 2		
1. Row number	1-3	I3
2. Row name	7-18	3A4

Control Card Format, SIZE

Control Cards SIZE and OPT have the appropriate numbers right justified in fields of five spaces each. The first card, SIZE, contains four numbers relating to the economic matrix being used for analysis. These numbers are the size of the transaction table (NN), the number of the last row of the processing sector (NP), the number of the last row of the value added sector (NV), and the number of the last row of the import sector (NI).

Item	Card Column	Format
1. NN	1-5	I5
2. NP	6-10	I5
3. NV	11-15	I5
4. NI	16-20	I5

Control Card Format, OPT

The option card contains three numbers. The first (MN) is for the environmental analysis. If environmental multipliers are being calculated MN = the number of environmental categories (positive & negative) that are being used. If no environmental analysis is desired MN = 0. The second number (LAB) indicates the length of the original (i.e., noncompressed) employment vector. If no employment data is present LAB = 0. The last number (NCOMP) indicates whether the matrix to be analyzed is a compressed version of the data being read in. If compression is desired, NCOMP = the number of times that several sectors are being compressed into one. If subroutine compress is not required, then NCOMP = 0.

Option Card Format

Item	Card Column	Format
1. Environmental Option (MN) MN = 0 deletes option MN = number of environmental sectors calls option	1-5	I5
2. Employment option (LAB) LAB = 0 deletes option LAB = number of processing sectors calls option	6-10	I5
3. Compression Option (NCOMP) NCOMP = 0 deletes option NCOMP = number of compressions within matrix calls option	11-15	I5

If NCOMP $\neq$ 0, the size of the matrix being used for analysis is smaller than the matrix being read in. Therefore, the data on card SIZE (NN, NP, NV, NI) refer to the compressed matrix and are smaller than they would be if compression were not desired.

Control Card Format, MOVE

The last set of data cards (deck MOVE) provide the information needed by the compression subroutine. The information consists of the size of the transaction matrix being read (since compress is being called, this number is larger than its counterpart on control card SIZE) and a vector of numbers in groups of three which states the destination, origin, and range of each compression.

Only adjacent sectors can be aggregated, and while some column and row numbers will change, the sequence must be maintained. The first group of sectors which are aggregated are placed in the row and column which the first sector of that group occupied. The unaggregated sectors between the first and second compression will automatically be moved over, and the sum of the second group

of aggregated sectors will be placed in the following row and column. The information presented in the compression vector, then, is (1) the number of the new column (row) which contains the aggregated sectors of the initial compression, (2) the number of the first sector in the group to be added (i.e., the starting position of the compression) and (3) the number of the last sector of the group (i.e., the end of that particular compression). Similar information is given for each compression. One compression includes summation of both columns and rows.

The data for subroutine COMPRESS are right justified in adjacent fields, containing five columns each. The first field begins in column one. The first card contains the size of the transaction matrix being read in and as many as four groups of compression data. If more than four compressions are desired the information is coded on subsequent cards as needed. All additional cards contain only the four groups of three numbers.

Move Format

Card Item	Card Column	Format
Card 1		
1. Dimension of matrix read in Dimension = N for an NxN matrix	1-5	I5
2. Column (Row) number of position of aggregate vector	6-10	I5
3. First column (Row) of group to be aggregated	11-15	I5
4. Last column (Row) of group to be aggregated	16-20	I5
5. Column (Row) number of position of aggregate vector	21-25	I5

Move Format (continued)

Card Item	Card Column	Format
6. First column (Row of group to be aggregated	26-30	I-5
⋮	⋮	⋮
13. Last column (Row) of group to be aggregated	61-65	I-5

Card 2

1. Column (Row) number of position of aggregated vector	I-5	I-5
2. First column (Row) of group to be aggregated	6-10	I-5

When COMPRESS is used the row names in NAME correspond to the aggregated sectors rather than the original sectors.

Example Code Sheets

Examples of the coding procedures are presented in the following code sheets. The data used are from a problem run as an illustration of the program's use and output.

DATA PROCESSING CENTER

TEXAS A & M UNIVERSITY

PROBLEM: DATA for ECON-ECOL

PROGRAMMER:

PAGE OF

DATE

FORTRAN STATEMENT

C COMMENT | 5 | 6 CONTINUATION | 7 | 10 | 20 | 30 | 40 | 50 | 60

Economic data is coded as follows:

FORMAT (4(2I3, F12.3))

```
 1  1      100.000   1  2       50.000   1  4      500.000   1  5        20
 1  6      500.000   1  7      500.000   1  8      650.000   2  2         7
```

Row Column Amount | Row Column Amount | Row Column Amount | Row Column Amo

Set 1 Set 2 Set 3 Set 4

Continue until all data for ECON is coded.

End ECON data with one card coded as follows:

```
-99
```

Environmental data is coded as follows:

FORMAT (4(2I3, F12.3))

```
 1  1     -500.000   1  2    -2000.000   1  4     -800.000   1  6        20
 3  1     1700.000   3  2     1600.000   3  3     1100.000   3  5        50
```

Row Column Amount | Row Column Amount | Row Column Amount | Row Column Amo

Set 1 Set 2 Set 3 Set 4

Continue until all data for RES is coded.

End RES data with one card coded as follows:

```
-99
```

Employment data is coded as follows:

FORMAT (6F10.0)

```
   476       200       400        25       300
```

Continue until all data for EMPLOY is coded

No special card is needed to end this section.

C COMMENT | CONTINUATION

DATA PROCESSING CENTER

TEXAS A & M UNIVERSITY

PROBLEM: DATA for ECON-ECOL

PROGRAMMER:

PAGE

DATE

FORTRAN STATEMENT

Row names are coded as follows:

FORMAT (I3, 3X, 3A4)

```
1
2  AGRICULTURE
   MINING
```

Row | Blank | Row Name

0

Continue until all data for NAME is coded.

No special card is needed to end this section.

Control cards are coded as follows:

FORMAT (6I5)

SIZE (1 card)

```
   6    3    5    6
```

NN | NP | NV | NI

OPT (1 card)

```
   3    3    2
```

MN | EMPLOY | NCOMP

Compression data is coded as follows:

FORMAT (13I5/, 12I5)

```
   8    1    1    2    3    4    5
```

NO | Destination Start | End | Destination Start | End

Continue until all data for MOVE is coded.

No special card is needed to end this section.

Complete Job Stream

Control cards for the computer system consist of a job card, a class card, and supporting JCL cards. At maximum a computer run with this program should not exceed 3 minutes, 5000 lines, or require more than 320K for storage space. It is possible that for smaller data sets smaller parameters may be desired in order to achieve faster turn around.

The program is written for Fortran G. Due to core requirements it cannot be run on the special superfast compilers such as WATFIV. JCL cards used with the program are indicated in the complete job stream (figure 3).

Output

The results of this program contain matrices presenting standard results of input-output analysis (i.e., transaction matrix, direct coefficients, and inverses with and without households), vectors of income and employment multipliers, and vectors of environmental multipliers.

The environmental multipliers present the trade-offs between the environmental factors and output, income, and employment. In addition, resource-resource multipliers are presented which show the total amount of resource use in the region for a unit increase in its usage by each industry.

With the exception of transaction table all output is self-explanatory. A possible source of confusion is the numbered columns of this matrix. Within the processing section the columns correspond to the named rows. However, since final demand categories can be different from final payments, the column designations must be kept track of separately by the analyst. Finally the last two columns contain respectively sub-totals of the processing sectors and the total of the entire row.

COMPLETE JOB STREAM

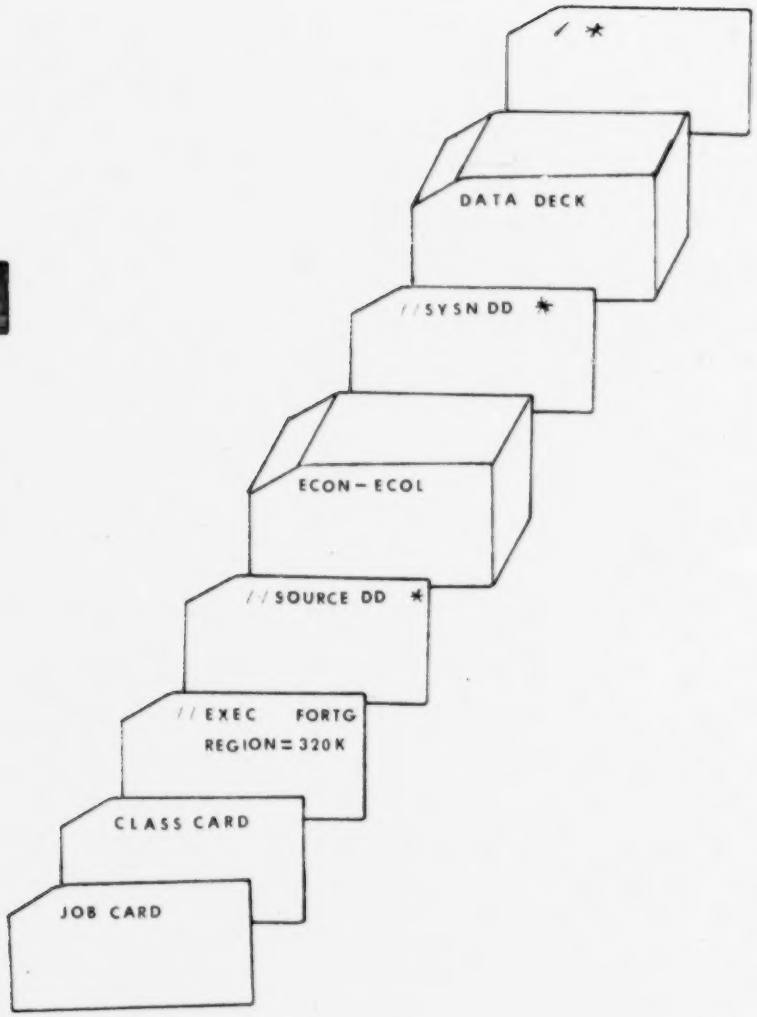


FIGURE 3

Example Problem

The following is output from a sample set of data. The data is from an 8 x 8 transaction matrix which has been reduced, via COMPRESS, to 6 x 6. It should be noted that it is impossible to get output of data before and after compression on the same computer run (i.e., NCOMP cannot = 0 and ≠ 0 at the same time). What follows is the combination of two runs for example purposes.

The environmental factors consists of two pollutants and one resource.

ORIGINAL ECON DATA

	1	2	3	4
1.AGRICULTURE	100.000	50.000	0.0	500.000
2. MINES	0.0	75.000	225.000	200.000
3.CONST-MANUF	200.000	200.000	100.000	0.0
4.UTILITIES	100.000	0.0	300.000	250.000
5.COMMERCE	200.000	200.000	400.000	500.000
6.HOUSEHOLDS	300.000	500.000	275.000	250.000
7.GOVERNMENT	100.000	275.000	100.000	100.000
8.IMPORTS	200.000	300.000	0.0	300.000

	5	6	7	8
1.AGRICULTURE	200.000	500.000	500.000	450.000
2. MINES	300.000	750.000	0.0	250.000
3.CONST-MANUF	800.000	100.000	0.0	0.0
4.UTILITIES	500.000	400.000	0.0	350.000
5.COMMERCE	100.000	400.000	400.000	0.0
6.HOUSEHOLDS	400.000	0.0	225.000	0.0
7.GOVERNMENT	0.0	0.0	0.0	0.0
8.IMPORTS	0.0	300.000	0.0	0.0

TRANSACTION TABLE COMPRESSED OUTPUT

	1	2	3	4
1.AGRI-MINES	225.00	225.00	1200.00	1250.00
2.CONST-MANUF	400.00	100.00	800.00	100.00
3.UTIL-COMM	800.00	700.00	1350.00	800.00
INT. PURCHASES	1425.00	1025.00	3350.00	2150.00
4.HOUSEHOLDS	1300.00	275.00	650.00	0.0
5.GOVERNMENT	1075.00	100.00	100.00	0.0
VALUE CREATED	2375.00	375.00	750.00	0.0
6.IMPORTS	500.00	0.0	300.00	300.00
TOTAL IMPORTS	500.00	0.0	300.00	300.00
COLUMN TOTALS	4300.00	1400.00	4400.00	2450.00

TRANSACTION TABLE (continued)

	5	6	7 (Sub-total)	8 (Total)
1.AGRI-MINES	500.00	900.00	1650.00	4300.00
2.CONST-MANUF	0.0	0.0	1300.00	1400.00
3.UTIL-COMM	400.00	350.00	2850.00	4430.00
INT. PURCHASES	900.00	1250.00	5800.00	10100.00
4.HOUSEHOLDS	225.00	0.0	2225.00	2450.00
5.GOVERNMENT	0.0	0.0	1275.00	1275.00
VALUE CREATED	225.00	0.0	3500.00	3725.00
6.IMPORTS	0.0	0.0	800.00	1100.00
TOTAL IMPORTS	0.0	0.0	800.00	1100.00
COLUMN TOTALS	1125.00	1250.00	0.0	0.0

ROW SUM = 14925.0000 COLUMN SUM = 14925.0000

ENVIRONMENTAL FACTORS

	1	2	3	4
1. SULFIDES	-2500.00	0.0	-800.00	-200.00
2. BOD	-300.00	-100.00	-425.00	0.0
3. WATER REQ	3300.00	1100.00	500.00	2000.00

EMPLOYMENT VECTOR

1.	675.00
2.	400.00
3.	325.00

DIRECT REQUIREMENTS (Econ technical coefficients)

	1	2	3
1. AGRI-MINES	0.05233	0.16071	0.27273
2. CONST-MANUF	0.09302	0.07143	0.18182
3. UTIL-COMM	0.18605	0.50000	0.30682
4. HOUSEHOLDS	0.30233	0.19643	0.14773
5. GOVERNMENT	0.25000	0.07143	0.02273
6. IMPORTS	0.11628	0.0	0.06818
COLUMN TOTALS	1.00000	1.00000	1.00000

INTERDEPENDENCE TABLE

	1	2	3
1. AGRI-MINES	1.23319	0.55276	0.63018
2. CONST-MANUF	0.21932	1.35235	0.44100
3. UTIL-COMM	0.48918	1.12382	1.92986
COLUMN TOTALS	1.94169	3.02893	3.00104

DIRECT REQUIREMENTS (Econ technical coefficients)

	1	2	3	4
1.AGRI-MINES	0.05233	0.16071	0.27273	0.51020
2.CONST-MANUF	0.09302	0.07143	0.18182	0.04082
3.UTIL-COMM	0.18605	0.50000	0.30682	0.32653
4.HOUSEHOLDS	0.30233	0.19643	0.14773	0.0
5.GOVERNMENT	0.25000	0.07143	0.02273	0.0
6.IMPORTS	0.11628	0.0	0.06818	0 12245
COLUMN TOTALS	1.00000	1.00000	1.00000	1.00000

INTERDEPENDENCE TABLE

	1	2	3	4
1.AGRI-MINES	2.00425	1.49851	1.51822	1 57948
2.CONST-MANUF	0.49906	1.69546	0.76318	0.57303
3.UTIL-COMM	1.32148	2.14468	2.88843	1.70492
4.HOUSEHOLDS	0.89918	1.10290	1.03561	1.84194
COLUMN TOTALS	4.72396	6.44155	6.20544	5.69937

	INCOME MULTIPLIER TYPE I
1	1.6147
2	3 0483
3	3.8059

	INCOME MULTIPLIER TYPE II
1	2.9742
2	5.6148
3	7 0103
4	1.8419

EMPLOYMENT MULTIPLIERS

	DIRECT EFFECT	TOTAL EFFECT	MULTIPLIER
1	0.15698	0 29238	1.86256
2	0.28571	0.55616	1.94658
3	0 07386	0.36747	4.97499

ENVIR. FACTORS PER DOLLAR OUTPUT

	1	2	3	4
1.SULFIDES	-0.581395	0.0	-0.181918	-0.091633
2.BOD	-0.069767	-0.071429	-0.096591	0.0
3.WATER REQ	0.767442	0.785714	0.113636	0.816326

ENVIRONMENTAL INTERDEPENDENCE MATRIX

	1	2	3
1.SULFIDES	-0.805911	-0.525704	-0.717265
2.BOD	-0.148953	-0.243712	-0.261873
3.WATER REQ	1.174313	1.614474	1.049428

ENV. SELF MULTIPLIER I

	1	2	3
1.SULFIDES	1.386167**********		3.944956
2.BOD	2.134987	3.411963	2.711154
3.WATER REQ	1.530165	2.054785	9.234970

ENVIRONMENTAL INTERDEPENDENCE MATRIX II

	1	2	3	4
1.SULFIDES	-1.478930	-1.351203	-1.492393	-1.278453
2.BOD	-0.303120	-0.432808	-0.439431	-0.315306
3.WATER REQ	2.814453	3.626207	2.938409	3.350757

ENV. SELF MULTIPLIER II

	1	2	3	4
1.SULFIDES	2.543758**********		8.208161	16.884480
2.BOD	4.344728	6.059314	4.549405**********	
3.WATER REQ	3.667317	4.615172	25.858002	4.115703

ENVIRON - EMPLOYMENT MULTIPLIERS

	1	2	3
1.SULFIDES	-2.756400	-0.945231	-1.951896
2.BOD	-0.509452	-0.438201	-0.712636
3.WATER REQ	4.016418	2.902871	2.855915

TYPE I ENVIRON - INCOME MULTIPLIERS

	1	2	3
1.SULFIDES	-1.650880	-0.877970	-1.275733
2.BOD	-0.305124	-0.407019	-0.465770
3.WATER REQ	2.405537	2.696308	1.866523

TYPE II ENVIRON - INCOME MULTIPIERS

	1	2	3	4
1.SULFIDES	-1.644750	-1.225133	-1.441080	-0.748479
2.BOD	-0.337107	-0.392426	-0.424322	-0.171453
3.WATER REQ	3.130014	3.287876	2.837379	1.824034

PROGRAMMER DOCUMENTATION

The Input-Output model used in this program presents the economy of a particular region as a system of simultaneous linear equations. These equations represent the distribution of each economic sector's output among its purchasers. Purchasers are separated into two main groups, the processing sector and final demand. The system of equations is solved to give the output of industries of the processing sector expressed as a function of the final demand sector. The solution is in the form of coefficients that express the total change in output in each industry for a change in final demand in each industry.

The environmental matrix is formed by expressing resource use in terms of units per dollar of output. When this matrix is multiplied times the above matrix of coefficients, the total change in environmental factors (positive for resource inputs, negative for pollutants) per dollar change in final demand are estimated. Trade-offs between environmental factors and economic variables such as income and employment are estimated by combining relations found in the economic analysis with those derived in the environmental analysis.

Basic Mathematical Model

The following is a mathematical presentation of the model.

$$\text{Let, } X_i = x_{ij} + \ldots x_{in} + y_i \qquad i = 1,2,\ldots n \qquad (1)$$

where X_i = total output of industry i

x_{ij} = purchase of output from industry i by industry j

y_i = purchase of output from industry i by final demand sectors.

It is assumed that a linear relationship exists between purchases of a sector from other sectors and the level of output by that sector. Since total output equals total purchases, this may be expressed as:

$$x_{ij} = a_{ij}X_j \tag{2}$$

or

$$a_{ij} = x_{ij}/X_j \tag{3}$$

where, X_j = total purchases by sector j.

Substituting (2) into (1) yields the equation,

$$X_i = a_{ij}X_1 + a_{i2}X_2 + \ldots + a_{in}X_n + y_i \tag{4}$$

Data being fed into the program consists of purchases, x_{ij} of equation (1), and is printed out in the data and transactions matrix. The a_{ij} elements of equation (4) are calculated and presented as elements of the Direct Requirements matrix, sometimes referred to as the technical coefficients of production.

By rearranging (4) this system of equation may be written,

$$\begin{aligned}
(1 - a_{11})X_1 &- a_{12}X_2 - a_{13}X_3 \quad \ldots - a_{1n}X_n = y_1 \\
a_{21}X_1 &- (1 - a_{22})X_2 - a_{23}X_3 \quad \ldots - a_{2n}X_n = y_2 \\
&\vdots \\
a_{n1}X_1 &- a_{n2}X_2 - a_{n3}X_3 \ldots - (1 - a_{nn})X_n = y_n
\end{aligned} \tag{5}$$

Equation (5) can then be expressed in matrix notation as follows:

$$(I - A)X = Y \tag{6}$$

Inverting the matrix $(I - A)$ yields

$$X = (I - A)^{-1}Y \tag{7}$$

The $(I - A)^{-1}$ represents the matrix of partial derivatives of X with respect to Y (i.e., $\partial X_i/\partial y_j$). Elements of this matrix are commonly referred to as the matrix of interdependence coefficients and represent the change in output of

industry i for a change in final demand in industry j.

The environmental data is transformed into resource use (pollution) per unit of output. The relationship of resource use (pollution) and output is assumed to be a linear function also,

$$r_{kj} = b_{kj}X_j \tag{8}$$

where

r_{kj} = amount resource requirement or pollution output

X_j = output of sector j

and b_{kj} = a constant.

The technical coefficients b_{kj} are thus derived from the combination of calculated output and collected resource data as follows,

$$b_{kj} = r_{kj}/X_j . \tag{9}$$

Let R_k = total regional level (positive or negative) of environmental factor K. Thus,

$$\begin{aligned} R_1 &= b_{11}X_1 + b_{12}X_2 + \ldots + b_{in}X_n \\ R_2 &= b_{21}X_1 + b_{22}X_2 + \ldots + b_{2n}X_n \\ &\vdots \qquad\qquad\qquad\qquad \vdots \\ R_m &= b_{m1}X_1 + b_{m2}X_2 + \ldots + b_{mn}X_n \end{aligned} \tag{10}$$

Again, in matrix notation

$$R = BX$$

where $R = \begin{bmatrix} R_1 \\ R_2 \\ \vdots \\ R_m \end{bmatrix}$ $\quad B = \begin{bmatrix} b_{11} & b_{12}\cdots b_{1n} \\ b_{21} & b_{22}\cdots b_{2n} \\ \vdots & \vdots \quad \vdots \\ b_{m1} & b_{m2} \quad b_{mn} \end{bmatrix}$ $\quad X = \begin{bmatrix} X_1 \\ X_2 \\ \\ X_n \end{bmatrix}$

Substituting equation (7) from the input-output model into equation (11) yields the environmental-output equation,

$$R = B(I - A)^{-1}Y. \tag{12}$$

For simplicity, let $P = B(I - A)^{-1}Y$ so that equation (12) becomes

$$R = P(Y). \tag{13}$$

The P matrix contains the environmental-output multipliers which express the relation between a change in final demand in a given sector and the total change in a given environmental factor. For resource use (e.g., water, minerals, land) these multipliers are positive and may be interpreted as the total change in the use of the resource within the region per dollar of final demand in a particular sector. For pollutants the multipliers are negative, but their interpretation is the same. A P_{kj} element indicates the total change in environmental factor k per dollar change in final demand in industry j.

Calculation of Multipliers

Environmental-Income Multipliers

The trade-off between environmental factors and income is found by using the total income effect and the resource-output multiplier. The total income effect is usually found as the first step in generating income multipliers. If households are exogenous to the processing sector then the total income effect is

$$(9) \quad Z = W(I - A)^{-1}$$

where Z = a vector of multipliers indicating the change in wages in a sector for a dollar change in output for that sector ($\partial W/\partial y$).

W = the vector of wages paid per dollar of output.

The type I environmental-income multipliers are then found by dividing each environmental-output multiplier by the total income effect of the appropriate sector.

$$R_Z = {}^{P_{kj}}/_{Z_j}$$

where Z_j = the jth element of vector Z.

R_Z = matrix of environmental-income multipliers

When the households sector is made endogenous to the processing sector, the procedure is similar to the above. However, the vector Z is calculated directly in the inversion process that yields the interdependence coefficients. By using the household row of the inverted matrix, we obtain the type II resource income multipliers. The difference between type I multiplier as described above and type II multiplier is that the induced spending of households is included in the latter.

Environmental-Employment Multipliers

Multipliers for environmental factors and employment interrelations are calculated in the same manner as the type I environmental-income multipliers. In place of the income vector Z, a vector of employment coefficients is used. These coefficients are found by dividing the total employment of a sector by that sector's total output. The resulting vector will then contain elements which are the average number of employees per dollar of output. Total change in employment for the economy is then estimated by the equation,

$$(10) \quad E = L(I - A)^{-1}$$

where E = the total effect on employment for a dollar change in each industry's output

L = the vector of direct employment coefficients

Then, environmental-employment interrelations are calculated from,

$$(11) \quad R_E = {}^{P_{kj}}/_{E_j}$$

where R_E = a matrix of multipliers indicating the change in environmental factor usage for a unit change in each industry's employment.

Environmental-Self Multipliers

The relationship between the amount of environmental factors an industry uses (or produces in the case of pollutants) and that which it causes to be used through its purchases of inputs may be similarily estimated from the model. This "environmental-self" multiplier is calculated by dividing the environmental output multiplier for a particular economic sector and environmental category by the corresponding sector's environmental factor use per dollar of output.

That is

$$(12) \quad R_R = {}^{P_{kj}}/_{b_{kj}}$$

where R_R = a matrix of environmental multipliers relating the total change in regional use of factor k for each unit change in its use in sector j.

PROGRAM DESCRIPTION

The program is written in FORTRAN IV with single precision. The JCL cards presented in the user documentation are for a one step execution available on the Texas A&M University data processing system. Other systems may require a two or three step execution and corresponding JCL.

Routines

MAIN: This routine reads data in, makes the calculations demonstrated in the mathematical model, and prints out the results.

INVERT: This subroutine uses Gaussian Elimination to invert the (I-A) matrix. The (I-A) matrix is placed in storage as DATA and passed to INVERT along with supporting variables and vectors.

Information passed	Received as
DATA	X
L	N
IER	IERR
LL	L
MM	M

COMPRESS: Aggregation of collected data is done by this subroutine as an option. The main routine then does its calculations with the reduced matrices and vectors. Four variables and one vector are passed to COMPRESS. Two matrices and one vector are placed in common with MAIN and COMPRESS and are accessed through BLOCK DATA.

Information Passed	Received As
NC	NC
NN	NEW
NO	NO
MN	MN
NVEC	NV

BLOCK DATA: holds matrices DATA and POL and vector EMP in common with MAIN and COMPRESS. DATA contains the $(I-A)^{-1}$ matrix, POL contains the environmental factors, and EMP is the vector of employment totals.

Dimensions and Initial Data Statements

MAIN

The maximum size of the needed matrices and vectors are specified in the dimension statements, and selected ones are initially filled with zeroes by data statements. Matrices DATA and POL and vector EMP are placed on a common card for access by COMPRESS.

INVERT

Matrix X and vectors L and M are dimensioned initially for use in this subroutine.

COMPRESS

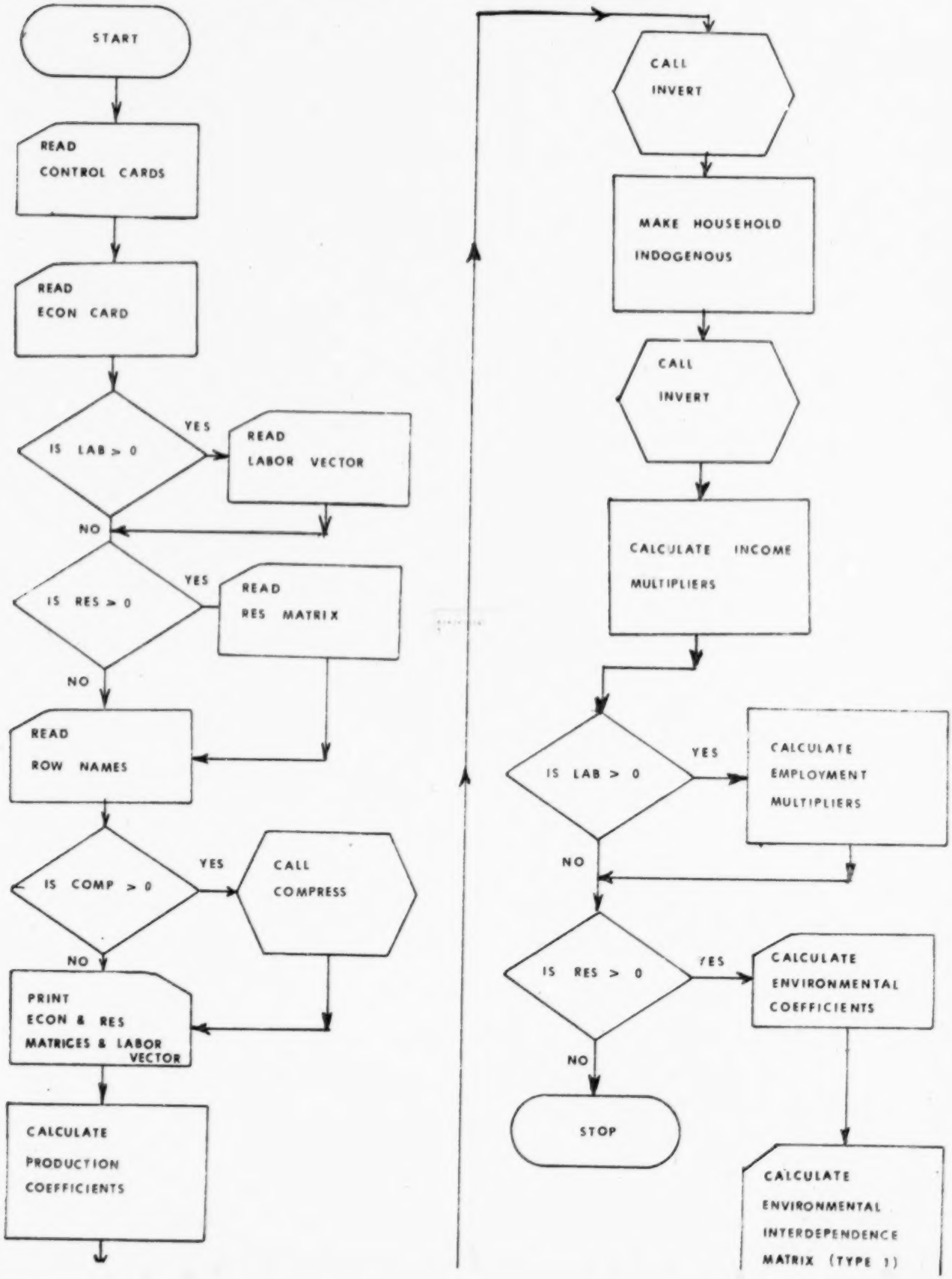
DATA is dimensioned as DATIX. The maximum size of matrix POL and vectors NV and EMP are given. A common statement connects DATIX, EMP, and POL with MAIN.

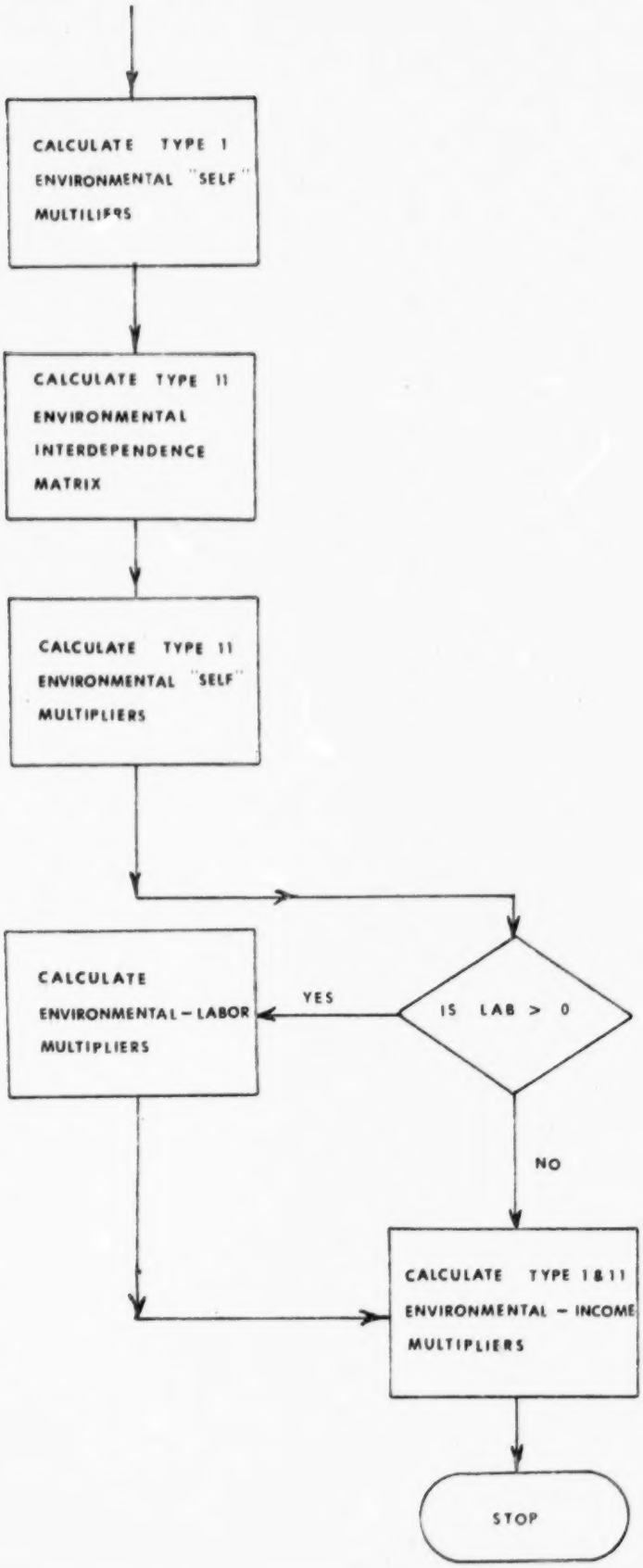
BLOCK DATA

The arrays held in common, DATA, POL, and EMP, are dimensioned and filled with zeroes.

Should it be desired to increase or decrease the maximum size of any matrix or vector, care should be taken to see that all dimension cards containing the array have been properly altered. Also that, if applicable, the proper number of zeroes are placed in the data statement.

FLOW CHART


START
READ CONTROL CARDS
READ ECON CARD
IS LAB > 0
YES
READ LABOR VECTOR
NO
IS RES > 0
YES
READ RES MATRIX
NO
READ ROW NAMES
IS COMP > 0
YES
CALL COMPRESS
NO
PRINT ECON & RES MATRICES & LABOR VECTOR
CALCULATE PRODUCTION COEFFICIENTS
CALL INVERT
MAKE HOUSEHOLD INDOGENOUS
CALL INVERT
CALCULATE INCOME MULTIPLIERS
IS LAB > 0
YES
CALCULATE EMPLOYMENT MULTIPLIERS
NO
IS RES > 0
YES
CALCULATE ENVIRONMENTAL COEFFICIENTS
NO
STOP
CALCULATE ENVIRONMENTAL INTERDEPENDENCE MATRIX (TYPE 1)


CALCULATE TYPE I ENVIRONMENTAL "SELF" MULTILIERS
CALCULATE TYPE II ENVIRONMENTAL INTERDEPENDENCE MATRIX
CALCULATE TYPE II ENVIRONMENTAL "SELF" MULTIPLIERS
IS LAB > 0
YES
CALCULATE ENVIRONMENTAL – LABOR MULTIPLIERS
NO
CALCULATE TYPE I & II ENVIRONMENTAL – INCOME MULTIPLIERS
STOP

PROGRAM LISTING

```
0001                  REAL MUL1,MUL2
0002                  DIMENSION  DATA(100,100),EMP(100),POL(25,100),
                     1NAME(100,4),NAM2(25,4),CTOT(100),RTOT(100),
                     2DINT(90,90),PINT(25,90),DTEC(90,90),STOR(25,95),
                     3MM(100),NVEC(95),NAME2(1,4),STOR2(100),MUL1(100),
                     4II(4),JJ(4),LL(100),DAT2(95,95),MUL2(100),X(4)
0003                  COMMON /DEP/ DATA,EMP,POL
0004                  DATA II/4*0/,JJ/4*0/,X/4*0.0/,NVEC/95*0/
0005                  DATA IER/0/
                C     ************************
                C     *  CONTROL  CARD READ  *
                C     ************************
0006                  READ (5,5) NN,NP,NV,NI
0007                  READ (5,5) MN,LAB,NCOMP
0008            5     FORMAT (6I5)
0009                  MNA1=MN+1
0010                  NVA1=NV+1
0011                  NNA1=NN+1
0012                  NNA2=NN+2
0013                  NNA3=NN+3
0014                  NNA4=NN+4
0015                  NNA5=NN+5
0016                  NPA1=NP+1
0017                  NPA2=NP+2
0018                  NPA3=NP+3
                C
                C     *******************************
                C     *   DATA MATRIX IS READ IN    *
                C     *******************************
                C
0019           10     READ (5,15) (II(I),JJ(I),X(I),I=1,4)
0020           15     FORMAT(4(2I3,F12.3))
0021                  IF (II(1).EQ.-99) GO TO 25
0022                  DO 20 J=1,4
0023                  K=II(J)
0024                  L=JJ(J)
0025                  IF (K.EQ.0.OR.L.EQ.0) GO TO 20
0026                  DATA (K,L)=X(J)
0027           20     CONTINUE
0028                  GO TO 10
                C
                C     *******************************
                C     * EMPLOYMENT VECTOR READ      *
                C     *******************************
                C
0029           25     IF (LAB.EQ.0) GO TO 35
0030                  READ (5,30) (EMP(I),I=1,LAB)
0031           30     FORMAT (6F10.0)
                C
                C
                C     ***********************************
                C     *    RES    MATRIX IS READ IN     *
                C     ***********************************
                C
0032           35     IF (MN.EQ.0) GO TO 55
0033           40     READ (5,45) (II(I),JJ(I),X(I),I=1,4)
0034           45     FORMAT (4(2I3,F12.2))
0035                  IF (II(1).EQ.-99) GO TO 55
```

```
0036                DO 50 J=1,4
0037                IF (II(J).EQ.0.OR.JJ(J).EQ.0) GO TO 50
0038                K=II(J)
0039                L=JJ(J)
0040                POL(K,L)=X(J)
0041          50    CONTINUE
0042                GO TO 40
                C
                C   ****************************
                C   * READ REGIONS AND TITLES *
                C   ****************************
                C
0043          55    DO 60 I=1,NN
0044          60    READ (5,65) (NAME(I,J),J=1,4)
0045          65    FORMAT (I3,3X,3A4)
0046                IF (MN.EQ.0) GO TO 75
0047                DO 70 I=1,MN
0048          70    READ (5,65) (NAM2(I,J),J=1,4)
0049          75    CONTINUE
                C
                C   ********************************
                C   *  CALL SUBROUTINE COMPRESS  *
                C   ********************************
                C
0050                IF (NCOMP.EQ.0) GO TO 85
0051                NC=NCOMP*3
0052                READ (5,80) NO,(NVEC(I),I=1,NC)
0053          80    FORMAT (13I5/,(12I5))
0054                CALL CMPRSS (NC,NN,NO,MN,NVEC)
0055          85    CONTINUE
                C
                C   ****************************
                C   * DATA MATRIX IS PRINTED *
                C   ****************************
                C
0056                DO 95 I=1,NN,9
0057                K=I+8
0058                IF (K.GE.NN) K=NN
0059                WRITE (6,90) (L,L=I,K)
0060          90    FORMAT ('1',14X,9(10X,I2))
0061                DO 95 J=1,NN
0062          95    WRITE (6,100) (NAME(J,L),L=1,4),(DATA(J,N),N=I,K)
0063          100   FORMAT (I4,'.',3A4,2X,9F12.3)
                C
                C   ************************************************************
                C   * TOTALS & SUBTOTALS OF COLUMNS & ROWS                     *
                C   *         CALCULATE  PRINT  &                              *
                C   *                  STORE                                   *
                C   ************************************************************
                C
0064                SUMR=0.0
0065                SUMC=0.0
0066                DO 110 I=1,NN
0067                RTOT(I)=0.0
0068                DO 105 J=1,NN
0069          105   RTOT(I)=RTOT(I)+DATA(I,J)
0070          110   SUMR=SUMR+RTOT(I)
0071                DO 120 I=1,NN
```

```
0072                CTOT(I)=0.0
0073                DO 115 J=1,NN
0074          115   CTOT(I)=CTOT(I)+DATA(J,I)
0075          120   SUMC=SUMC+CTOT(I)
0076                DO 121 I=1,NN
0077                DO 121 J=1,NP
0078          121   DATA (I,NNA1)=DATA(I,NNA1)+DATA(I,J)
0079                DO 122 I=1,NN
0080          122   DATA (I,NNA2)=RTOT(I)
0081                DO 125 I=1,NNA2
0082                DO 123 J=1,NP
0083          123   DATA (I,NNA3)=DATA(I,NNA3)+DATA(J,I)
0084                DO 124 K=NPA1,NV
0085          124   DATA (I,NNA4)=DATA(I,NNA4)+DATA(K,I)
0086                DO 125 L=NVA1,NI
0087          125   DATA (I,NNA5)=DATA(I,NNA5)+DATA(L,I)
                    C
                    C     ************************************
                    C     * TRANSACTIONS MATRIX IS PRINTED *
                    C     ************************************
                    C
0088                DO 160 I=1,NNA2,9
0089                K=I+8
0090                IF (K.GT.NNA2) K=NNA2
0091                WRITE (6,126) (L,L=I,K)
0092                DO 150 J=1,NN
0093                WRITE (6,145) (NAME(J,L),L=1,4),(DATA(J,N),N=I,K)
0094                IF (J.EQ.NP) WRITE (6,130) (DATA(L,NNA3),L=I,K)
0095                IF (J.EQ.NV) WRITE (6,135) (DATA(L,NNA4),L=I,K)
0096                N1=NP
0097                IF (J.EQ.NI) WRITE (6,140) (DATA(L,NNA5),L=I,K)
0098                IF (J.EQ.NI) GO TO 150
0099          126   FORMAT('1 TRANACTION  TABLE'///' ',15X,9(10X,I2))
0100          130   FORMAT(//2X,'INT. PURCHASES ',3X,9F12.2//)
0101          135   FORMAT(//2X,'VALUE CREATED ',3X,9F12.2//)
0102          140   FORMAT(//2X,'TOTAL IMPORTS ',3X,9F12.2//)
0103          145   FORMAT (I4,'.',3A4,2X,9F12.2)
0104          150   CONTINUE
0105                WRITE (6,155) (CTOT(M4),M4=I,K)
0106          155   FORMAT(//2X,'COLUMN TOTALS',3X,9F12.2//)
0107          160   CONTINUE
0108                WRITE (6,165) SUMR,SUMC
0109          165   FORMAT('1',' ROW SUM = ',F17.4,5X,'COLUMN SUM =',F17.4)
0110                DO 170 I=1,NN
0111                DATA (NNA1,I)=CTOT(I)
0112          170   CONTINUE
                    C
                    C     ********************************************
                    C     * ENVIRONMENTAL MATRIX & EMPLOYMENT      *
                    C     *     VECTOR IS PRINTED                  *
                    C     ********************************************
                    C
0113                IF (MN.EQ.0) GO TO 185
0114                DO 180 I=1,NPA1,9
0115                K=I+8
0116                IF (K.GT.NPA1) K=NPA1
0117                WRITE (6,175) (L,L=I,K)
0118          175   FORMAT('1',3X,'ENVIRONMENTAL FACTORS',/////15X,
```

```
                19(10X,I2))
0119                 DO 180 J=1,MN
0120                 WRITE (6,145) (NAM2(J,L),L=1,4),(POL(J,N),N=I,K)
0121          180    CONTINUE
0122          185    CONTINUE
0123                 IF (LAB.EQ.0) GO TO 195
0124                 WRITE (6,190) (I,EMP(I),I=1,NP)
0125          190    FORMAT('1',5X,'EMPLOYMENT  VECTOR',//,(5X,I2,',',5X,
                    1F10.2))
0126          195    CONTINUE
              C      DATA  MATRIX  IS  STORED  AS  DAT2
              C
0127                 DO 200 I=1,NNA1
0128                 DO 200 J=1,NNA1
0129          200    DAT2(I,J)=DATA(I,J)
              C
              C      ***********************************
              C      * DIRECT REQUIREMENTS ( 1 &2)  *
              C      *       CALCULATE & PRINT      *
              C      ***********************************
              C
0130          205    CONTINUE
0131                 DO 210 I=1,N1
0132                 CTOT(I)=0.0
0133                 DO 210 J=1,NN
0134                 DATA (J,I)=DATA(J,I)/DATA(NNA1,I)
0135          210    CTOT(I)=CTOT(I)+DATA(J,I)
0136                 DO 230 I=1,NPA1,9
0137                 K=I+8
0138                 IF (K.GT.N1) K=N1
0139                 M2=K
0140                 WRITE (6,215) (L,L=I,K)
0141          215    FORMAT('1 DIRECT  REQUIREMENTS'/15X,9(10X,I2))
0142                 DO 220 J=1,NN
0143          220    WRITE (6,225) (NAME(J,L),L=1,4),(DATA(J,N),N=I,K)
0144          225    FORMAT (I4,'.',3A4,2X,9F12.5)
0145          230    WRITE (6,235) (CTOT(M4),M4=I,K)
0146          235    FORMAT(//2X,'COLUMN TOTALS',3X,9F12.5)
0147                 IF (N1.EQ.NP) GO TO 245
0148                 DO 240 I=1,N1
0149                 DO 240 J=1,N1
0150          240    DTEC(I,J)=DATA(I,J)
0151          245    CONTINUE
              C
              C      ****************************
              C      *  INVERT MATRICES 1 & 2   *
              C      ****************************
              C
0152                 L=N1
0153                 M=1
0154                 DO 246 I=1,N1
0155                 DATA (I,I)=1.0-DATA(I,I)
0156                 DO 246 J=1,N1
0157          246    IF(I.NE.J) DATA(I,J)= -DATA(I,J)
0158                 CALL INVERT (DATA,L,IER,LL,MM)
0159                 IF (IER.NE.0) STOP
0160                 DO 250 I=1,N1
0161                 CTOT(I)=0.0
```

```
0162                DO 250 J=1,N1
0163          250   CTOT(I)=CTOT(I)+DATA(J,I)
                    C
                    C     ************************************
                    C     *   PRINT & STORE INVERSES 1 & 2   *
                    C     ************************************
                    C
0164                IF (N1.EQ.NPA1) GO TO 260
0165                DO 255 I=1,N1
0166                DO 255 J=1,N1
0167          255   DINT(I,J)=DATA(I,J)
0168          260   CONTINUE
0169                DO 275 I=1,N1,9
0170                K=I+8
0171                IF (K.GT.N1) K=N1
0172                WRITE (6,265) (L,L=I,K)
0173          265   FORMAT('1 INTERDEPENDENCE  TABLE'/15X,9(10X,I2))
0174                DO 270 J=1,N1
0175          270   WRITE (6,225) (NAME(J,L),L=1,4),(DATA(J,M4),M4=I,K)
0176                WRITE (6,235) (CTOT(M4),M4=I,K)
0177          275   CONTINUE
0178                N1=N1+1
0179                IF (N1.EQ.NPA2) GO TO 280
0180                DO 276 I=1,NNA1
0181                DO 276 J=1,NNA1
0182          276   DATA (I,J)=DAT2(I,J)
0183                GO TO 205
                    C
                    C     ************************************************************
                    C     * AT THIS POINT THE STORED MATRICES ARE AS FOLLOWS         *
                    C     *   DINT=INVERSE I            DAT2=ORIG ECON DATA          *
                    C     *   DTEC=TECH COEFF W/ HH     POL =ORIG RES DATA           *
                    C     *   DATA=INVERSE II                                        *
                    C     ************************************************************
                    C
                    C     *********************************
                    C     *  INCOME MULTIPLIERS I &II     *
                    C     *********************************
                    C
0184          280   DO 290 I=1,NP
0185                VALUE=0.0
0186                DO 285 K=1,NP
0187          285   VALUE=VALUE+DINT(K,I)*DTEC(NPA1,K)
0188          290   STOR2(I)=VALUE
                    C                                          TYPE I
                    C                                    CALCULATE & STORE
0189                DO 295 I=1,NP
0190          295   MUL1(I)=(STOR2(I)/DTEC(NPA1,I))
                    C                                          TYPE II
                    C                                    CALCULATE & STORE
0191                DO 300 I=1,NPA1
0192          300   MUL2(I)=(DATA(NPA1,I)/DTEC(NPA1,I))
                    C                                                PRINT
                    C                                                I&II
0193                WRITE (6,305) (I,MUL1(I),I=1,NP)
0194          305   FORMAT('1',10X,'INCOME MULTIPLIER'/' ',15X,'TYPE I',
                   1/(6X,I2,6X,F9.4))
0195                WRITE (6,310) (I,MUL2(I),I=1,NPA1)
```

```
0196        310   FORMAT('1',10X,'INCOME MULTIPLIERS'//' ',15X,'TYPE II',
                 1//(5X,I2,6X,F9.4))
              C
              C     ***************************************************
              C     *   EMPLOYMENT MULTIPLIERS                        *
              C     * CALCULATE, PRINT, & STORE IN DATA(NNA4,I)       *
              C     ***************************************************
              C
0197                  IF (LAB.EQ.0) GO TO 340
0198                  DO 315 I=1,NP
0199        315   EMP(I)=EMP(I)/DAT2(NNA1,I)
0200                  DO 325 I=1,NP
0201                  CTOT(I)=0.0
0202                  VALUE=0.0
0203                  DO 320 K=1,NP
0204        320   VALUE=VALUE+DINT(K,I)*EMP(K)
0205        325   CTOT(I)=VALUE
0206                  DO 330 J=1,NP
0207        330   RTOT(J)=CTOT(J)/EMP(J)
0208                  WRITE (6,335) (I,EMP(I),CTOT(I),RTOT(I),I=1,NP)
0209        335   FORMAT('1',28X,'EMPLOYMENT MULTIPLIERS'///13X,
                 1'DIRECT EFFECT',9X,'TOTAL EFFECT',9X,'MULTIPLIER',
                 2/(10X,I2,3X,F9.5,13X,F9.5,10X,F9.5))
0210                  DO 336 I=1,NP
0211        336   DATA (NNA4,I)=RTOT(I)
0212        340   CONTINUE
              C
              C     ****************************************
              C     * ENVIRONMENTAL FACTORS PER $ OUTPUT   *
              C     *CALCULATE PRINT & STORE IN POL(J,I)   *
              C     ****************************************
              C
0213                  IF (MN.EQ.0) GO TO 525
0214                  DO 345 I=1,NPA1
0215                  DO 345 J=1,MN
0216        345   POL(J,I)=POL(J,I)/DAT2(NNA1,I)
0217                  DO 365 I=1,NPA1,9
0218                  K=I+8
0219                  IF (K.GT.NPA1) K=NPA1
0220                  IF (I/2*2.EQ.I) WRITE (6,350) (L,L=I,K)
0221                  IF (I/2*2.EQ.I) GO TO 360
0222        350   FORMAT (/////15X,9(10X,I2))
0223                  WRITE (6,355) (L,L=I,K)
0224        355   FORMAT('1ENVIR. FACTORS PER DOLLAR OUTPUT'///15X,
                 19(10X,I2))
0225        360   CONTINUE
0226                  DO 365 J=1,MN
0227        365   WRITE (6,390) (NAM2(J,L),L=1,4),(POL(J,L),L=I,K)
              C
              C     ***********************************************
              C     * ENVIRONMENTAL INTERDEPENDENCE MATRIX        *
              C     *       CALCULATE, PRINT,& STORE IN PINT      *
              C     ***********************************************
              C
0228                  DO 370 I=1,MN
0229                  DO 370 J=1,NP
0230                  PINT(I,J)=0
0231                  DO 370 K=1,NP
```

```
0232          370   PINT(I,J)=PINT(I,J)+POL(I,K)*DINT(K,J)
0233                DO 395 I=1,NP,9
0234                K=I+8
0235                IF (K.GT.NP) K=NP
0236                IF (I/2*2.EQ.I) WRITE (6,350) (L,L=I,K)
0237                IF (I/2*2.EQ.I) GO TO 380
0238                WRITE (6,375) (L,L=I,K)
0239          375   FORMAT('1 ENVIRONMENTAL INTERDEPENDENCE ',
                   1'MATRIX'///15X,9(10X,I2))
0240          380   CONTINUE
0241                DO 385 J=1,MN
0242          385   WRITE (6,390) (NAM2(J,L),L=1,4),(PINT(J,N),N=I,K)
0243          390   FORMAT (I4,'.',3A4,2X,9F12.6)
0244          395   CONTINUE
                    C
                    C     ****************************
                    C     *     ENVIR. MULTIPLIERS   *
                    C     *       CALCULATE & PRINT  *
                    C     ****************************
                    C
                    C                                        TYPE I
                    C                                  ENVIRONMENTAL MULTIPLIER
                    C
0245                DO 400 I=1,MN
0246                DO 400 J=1,NP
0247                IF (POL(I,J).NE.0.0) DAT2(I,J)=(PINT(I,J)/POL(I,J))
0248                IF (POL(I,J).EQ.0.0.AND.PINT(I,J).EQ.0.0) DAT2(I,J)=0.0
0249          400   IF (POL(I,J).EQ.0.0.AND.PINT(I,J).NE.0.0) DAT2(I,J)
                   1=1000000
0250                DO 415 I=1,NP,9
0251                K=I+8
0252                IF (K.GT.NP) K=NP
0253                IF (I/2*2.EQ.I) WRITE (6,350) (L,L=I,K)
0254                IF (I/2*2.EQ.I) GO TO 410
0255                WRITE (6,405) (L,L=I,K)
0256          405   FORMAT('1 ENV. SELF MULTIPLIER I '///15X,9(10X,I2))
0257          410   CONTINUE
0258                DO 415 J=1,MN
0259          415   WRITE (6,390) (NAM2(J,L),L=1,4),(DAT2(J,N),N=I,K)
                    C
                    C                                        TYPE II
                    C                                 ENVIRONMENTAL INTERDEPENDENCE
                    C                                   MATRIX  STORE IN DTEC
                    C
0260                DO 420 I=1,MN
0261                DO 420 J=1,NPA1
0262                DTEC(I,J)=0.
0263                DO 420 K=1,NPA1
0264          420   DTEC(I,J)=DTEC(I,J)+POL(I,K)*DATA(K,J)
0265                DO 440 I=1,NPA1,9
0266                K=I+8
0267                IF (K.GT.NPA1) K=NPA1
0268                IF (I/2*2.EQ.I) WRITE (6,350) (L,L=I,K)
0269                IF (I/2*2.EQ.I) GO TO 430
0270                WRITE (6,425) (L,L=I,K)
0271          425   FORMAT('1 ENVIRONMENTAL  INTERDEPENDENCE ',
                   1'MATRIX II'///15X,9(10X,I2))
0272          430   CONTINUE
0273                DO 435 J=1,MN
```

```
0274        435   WRITE (6,390) (NAM2(J,L),L=1,4),(DTEC(J,N),N=I,K)
0275        440   CONTINUE
                  C
                  C                                                TYPE II
                  C                                          ENVIRONMENTAL MULTIPLIER
                  C
0276              DO 445 I=1,MN
0277              DO 445 J=1,NPA1
0278              IF (POL(I,J).NE.0.0) POL(I,J)=DTEC(I,J)/POL(I,J)
0279              IF (POL(I,J).EQ.0.0.AND.DTEC(I,J).EQ.0.0) POL(I,J)=0.0
0280        445   IF (POL(I,J).EQ.0.0.AND.DTEC(I,J).NE.0.0) POL(I,J)=
                 11000000
0281              DO 460 I=1,NPA1,9
0282              K=I+8
0283              IF (K.GT.NPA1) K=NPA1
0284              IF (I/2*2.EQ.I) WRITE (6,350) (L,L=I,K)
0285              IF (I/2*2.EQ.I) GO TO 455
0286              WRITE (6,450) (L,L=I,K)
0287        450   FORMAT('1 ENV. SELF MULTIPLIER II'///15X,9(10X,I2))
0288        455   CONTINUE
0289              DO 460 J=1,MN
0290        460   WRITE (6,390) (NAM2(J,L),L=1,4),(POL(J,N),N=I,K)
                  C
                  C     **********************************
                  C     * COMBINATION MULTIPLIERS        *
                  C     *     ENVIRONMENTAL - EMPLOY     *
                  C     *     ENVIRONMENTAL - INCOME     *
                  C     **********************************
                  C
                  C                                   ENVIRONMENTAL - EMPLOYMENT
                  C                                          CALCULATE & PRINT
0291              IF (LAB.EQ.0) GO TO 485
0292              DO 465 J=1,NP
0293              DO 465 I=1,MN
0294        465   STOR(I,J)=PINT(I,J)/CTOT(J)
0295              DO 480 I=1,NP,9
0296              K=I+8
0297              IF (K.GT.NP) K=NP
0298              IF (I/2*2.EQ.I) WRITE (6,350) (L,L=I,K)
0299              IF (I/2*2.EQ.I) GO TO 475
0300              WRITE (6,470) (L,L=I,K)
0301        470   FORMAT('1 ENVIRON - EMPLOYMENT MULTIPLIERS'///15X,
                 19(10X,I2))
0302        475   CONTINUE
0303              DO 480 J=1,MN
0304        480   WRITE (6,390) (NAM2(J,L),L=1,4),(STOR(J,N),N=I,K)
0305        485   CONTINUE
                  C                                   ENVIRONMENTAL - INCOME I
                  C                                          CALCULATE & PRINT
0306              DO 490 J=1,NP
0307              DO 490 I=1,MN
0308        490   PINT(I,J)=PINT(I,J)/STOR2(J)
0309              DO 505 I=1,NP,9
0310              K=I+8
0311              IF (K.GT.NP) K=NP
0312              IF (I/2*2.EQ.I) WRITE (6,350) (L,L=I,K)
0313              IF (I/2*2.EQ.I) GO TO 500
0314              WRITE (6,495) (L,L=I,K)
```

```
0315        495  FORMAT('1 TYPE I ENVIRON - INCOME MULTIPLIERS'///1EX,
                19(10X,I2))
0316        500  CONTINUE
0317             DO 505 J=1,MN
0318        505  WRITE (6,390) (NAM2(J,L),L=1,4),(PINT(J,N),N=1,K)
                 C
                 C                                  ENVIRONMENTAL - INCOME II
                 C                                        CALCULATE & PRINT
0319             DO 510 J=1,NPA1
0320             DO 510 I=1,MN
0321        510  DTEC(I,J)=DTEC(I,J)/DATA(NPA1,J)
0322             DO 525 I=1,NPA1,8
0323             K=I+8
0324             IF (K.GT.NPA1) K=NPA1
0325             IF (I/2*2.EQ.I) WRITE (6,350) (L,L=I,K)
0326             IF (I/2*2.EQ.I) GO TO 520
0327             WRITE (6,515) (L,L=I,K)
0328        515  FORMAT('1 TYPE II ENVIRON - INCOME MULTIPIERS'///1EX,
                19(10X,I2))
0329        520  CONTINUE
0330             DO 525 J=1,MN
0331             WRITE (6,390) (NAM2(J,L),L=1,4),(DTEC(J,N),N=I,K)
0332        525  CONTINUE
0333             STOP
0334             END
```

```
0001            SUBROUTINE INVERT(X,N,IERR,L,M)
                C   REAL MATRIX INVERSION SUBROUTINE BY GAUSSIAN
                C   ELIMINATION
                C
                C   THE FOLLOWING CALLING SEQUENCE SHOULD BE USED TO ENTER
                C   THE FOLLOWING CALLING SEQUENCE SHOULD BE USED TO ENTER
                C   THIS SUBROUTINE .....    X
                C           CALL INVERT(X,N,IERR)
                C   WHERE
                C           X IS THE MATRIX TO BE INVERTED
                C           N IS THE ORDER OF X
                C           IERR IS AN ERROR FLAG DENOTING SUCCESSFUL OR
                C           NON-SUCCESSFUL INVERSION OF X
                C
                C
                C
0002            DIMENSION X(100,101),L(100),M(100)
                C   SEARCH FOR LARGEST ELEMENT IN X
0003            DO 80 K = 1,N
0004            L(K) = K
0005            M(K) = K
0006            XBIG = X(K,K)
0007            DO 20 I = K,N
0008            DO 20 J = K,N
0009            IF(ABS(XBIG).GE.ABS(X(I,J))) GO TO 20
0010            XBIG = X(I,J)
0011            L(K) = I
0012            M(K) = J
0013      20    CONTINUE
                C   NOW INTERCHANGE ROWS
0014            IROW = L(K)
0015            IF(L(K).LE.K) GO TO 35
0016            DO 30 I = 1,N
0017            WAIT = -X(K,I)
0018            X(K,I) = X(IROW,I)
0019      30    X(IROW,I) = WAIT
                C   NOW INTERCHANGE COLUMNS
0020      35    ICOL = M(K)
0021            IF(M(K).LE.K) GO TO 45
0022            DO 40 J = 1,N
0023            WAIT = -X(J,K)
0024            X(J,K) = X(J,ICOL)
0025      40    X(J,ICOL) = WAIT
                C   NOW DIVIDE COLUMN BY MINUS PIVOT
0026      45    DO 55 IC = 1,N
0027            IF(IC.EQ.K) GO TO 55
0028            X(IC,K) = X(IC,K)/(-X(K,K))
0029      55    CONTINUE
                C   NOW REDUCE MATRIX
0030            DO 65 I = 1,N
0031            DO 65 J = 1,N
0032            IF(I.EQ.K) GO TO 65
0033            IF(J.EQ.K) GO TO 65
0034            X(I,J) = X(I,K)*X(K,J) + X(I,J)
0035      65    CONTINUE
                C   NOW DIVIDE ROW BY PIVOT
0036            DO 75 JRO = 1,N
0037            IF(JRO.EQ.K) GO TO 75
```

```
0038                X(K,JRO) = X(K,JRO)/X(K,K)
0039             75 CONTINUE
                 C  CONTINUE PRODUCT OF PIVOTS AND REPLACE PIVOT BY
                 C  RECIPROCAL
0040                CALL OVERFL(J1)
0041                CALL DVCHK(J2)
0042                X(K,K) = 1.0/X(K,K)
0043                CALL DVCHK(J2)
0044                IF(J2.EQ.1) GO TO 600
0045                CALL OVERFL(J1)
0046                IF(J1.EQ.1) GO TO 600
                 C  NOW CONTINUE OVERALL OPERATION
0047             80 CONTINUE
                 C  NOW FOR FINAL ROW AND COLUMN INTERCHANGE
0048                K = N
0049            100 K = K - 1
0050                IF(K.LE.0) RETURN
0051                I = L(K)
0052                IF(I.LE.K) GO TO 120
0053                DO 110 J = 1,N
0054                WAIT = X(J,K)
0055                X(J,K) = -X(J,I)
0056            110 X(J,I) = WAIT
0057            120 J = M(K)
0058                IF(J.LE.K) GO TO 100
0059                DO 130 I = 1,N
0060                WAIT = X(K,I)
0061                X(K,I) = -X(J,I)
0062            130 X(J,I) = WAIT
0063                GO TO 100
0064            600 IERR = 1
0065                RETURN
0066                END
```

```
0001            SUBROUTINE CMPRSS(NC,NEW,NO,MN,NV)
0002            DIMENSION   DATIX(100,100),POL(25,100),NV(95),EMP(100)
0003            COMMON/DEP/DATIX,EMP,POL
0004            DO 100 IC = 1,NC,3
0005            INC = NV(IC)
0006            IB = NV(IC +1)
0007            IED = NV(IC +2)
0008            IMV = NV(IC +3)
0009            IED2 = NV(IC +4)
0010            DO 50 I = 1,NO
0011            SUM = 0.0
0012            SUMP = 0.0
0013            DO 40 J = IB,IED
0014            IF(MN.EQ.0.OR.MN.LT.I) GO TO 40
0015            SUMP = SUMP + POL(I,J)
0016         40 SUM = SUM + DATIX(I,J)
0017            IF(MN.EQ.0.OR.MN.LT.I) GO TO 50
0018            POL(I,INC) = SUMP
0019         50 DATIX(I,INC) = SUM
0020            IF(IED+1.EQ.IED2) GO TO 100
0021            IA = IED + 1
0022            IZ = IED2 - 1
0023            IF(IZ.LT.0) GO TO 100
0024            DO 80 I = 1,NO
0025            K = INC + 1
0026            DO 80 J = IA,IZ
0027            IF(MN.EQ.0.OR.MN.LT.I) GO TO 70
0028            POL(I,K) = POL(I,J)
0029         70 DATIX(I,K) = DATIX(I,J)
0030         80 K = K + 1
0031        100 CONTINUE
0032            IF(IED.EQ.NO) GO TO 25
0033            DO 20 I = 1,NO
0034            K = INC+1
0035            M = IED +1
0036            DO 20 J = M,NO
0037            IF(MN.EQ.0.OR.MN.LT.I) GO TO 19
0038            POL(I,K) = POL(I,J)
0039         19 DATIX(I,K) = DATIX(I,J)
0040         20 K = K + 1
0041         25 IZE = NEW + 1
0042            DO 30 I =1,NO
0043            DO 30 J= IZE,NO
0044            IF(MN.EQ.0.OR.MN.LT.I) GO TO 30
0045            POL(I,J) = 0.0
0046         30 DATIX(I,J) = 0.0
0047            DO 200 IR = 1,NC,3
0048            INR = NV(IR)
0049            IB  = NV(IR +1)
0050            IED = NV(IR +2)
0051            IMV = NV(IR +3)
0052            IED2 = NV(IR +4)
0053            DO 170 J = 1,NEW
0054            SUM = 0.0
0055            DO 150 I = IB,IED
0056        150 SUM = SUM + DATIX(I,J)
0057        170 DATIX(INR,J) = SUM
0058            IF(IED+1.EQ.IED2) GO TO 200
```

```
0059                IA = IED +1
0060                IZ = IED2 -1
0061                IF(IZ.LT.0)  GO TO 200
0062                DO 180 J = 1,NEW
0063                K = INR + 1
0064                DO 180 I = IA,IZ
0065                DATIX(K,J) = DATIX(I,J)
0066             180 K = K +1
0067             200 CONTINUE
0068                IF(IED.EQ.NO) GO TO 125
0069                DO 120 J = 1,NEW
0070                K = INR + 1
0071                M = IED + 1
0072                DO 120 I = M,NO
0073                DATIX(K,J) = DATIX(I,J)
0074          120   K = K+ 1
0075          125   IZE = NEW + 1
0076                DO 130 J = 1,NEW
0077                DO 130 I = IZE,NO
0078          130   DATIX(I,J) = 0.0
0079                DO 300 IV = 1,NC,3
0080                INC = NV(IV)
0081                IB  = NV(IV +1)
0082                IED = NV(IV +2)
0083                IMV = NV(IV +3)
0084                IED2= NV(IV +4)
0085                SUM =0.0
0086                DO 240 J = IB,IED
0087          240   SUM = SUM + EMP(J)
0088                EMP(INC) = SUM
0089                IF(IED+1.EQ.IED2) GO TO 300
0090                IA = IED + 1
0091                IZ = IED2 - 1
0092                IF(IZ.LT.0) GO TO 300
0093                K = INC + 1
0094                DO 280 J = IA,IZ
0095                EMP(K) = EMP(J)
0096          280   K = K + 1
0097          300   CONTINUE
0098                IF(IED.EQ.NO) GO TO 225
0099                K = INC + 1
0100                M = IED + 1
0101                DO 220 J = M,NO
0102                EMP(K) = EMP(J)
0103          220   K = K + 1
0104          225   IZE = NEW + 1
0105                DO 330 J = IZE,NO
0106          330   EMP(J) = 0.0
0107          500   CONTINUE
0108                RETURN
0109                END
```

```
0001                BLOCK DATA
0002                DIMENSION  DATA(100,100),POL(25,100),EMP(100)
0003                COMMON/DEP/DATA,EMP,POL
0004                DATA  DATA/10000*0.0/,POL/2500*0.0/,EMP/100*0.0/
0005                END
```

REFERENCES

[1] Blaylock, James E. and L.L. Jones, An Analysis of Economic and Environmental Interrelations in the Lower Rio Grande Region of Texas, Texas A&M Experiment Station, Texas A&M University, January, 1973.

[2] Canion, R.L. and L. Trock, Input-Output as a Method of Evaluation of the Economic Impact. Texas A&M University, College Station, Texas, Water Resources Institute, Technical Report No. 12, May, 1968.

[3] Isard, Walter, "Ecologic Analysis for Regional Development", Regional Science Association Papers, 21:79-99, 1967.

[4] Laurent, Eugene A. and James C. Hite, Economic-Ecologic Analysis in the Charleston Metropolitan Region: An Input-Output Study, Clemson University, Clemson, South Carolina, Water Resources Research Institute, Report No. 19, April, 1971.

[5] Martin, W.E. and H.O. Carter, A California Interindustry Analysis Emphasizing Agriculture (Part I and II), Giannini Foundation Res. Rep. 278, February, 1968.